

Introduction

© 李浩东 3190104890@zju.edu.cn

© 孙正男 3190102989@zju.edu.cn

Basic Information

- **Soft Machine Vision**
- Smart Factory Innovation Club of Zhejiang University
- Class locations: 紫金港月牙楼301
- Class time: Sunday 9:30 to 11:30
- Number of students enrolled: 42

Course Outline

- Image basics: Pixels, Colors, Image formats
- Image processing techniques: Filtering, Binarization, cutting, Morphological transformation, Scale and rotation transformation, Image gradient
- Image pattern recognition: Line and circle detection, Feature point detection, Edge detection
- Image pattern recognition: Blob detection, Feature point detection, Template matching
- Neural Network basics: Neuron structure, Multi-layer Perceptron, Handwritten digit recognition, Loss function
- Neural Network basics: Gradient Descent, Backpropagation
- Modern Neural Network: Softmax regression, Deep neural network, Convolutional Neural Network, Server Resources
- Modern Neural Network: Recurrent Neural Networks, Attention Mechanism & Transformer, Natural Language Processing, Reinforcement Learning, Generative Adversarial Networks
- Final project and Q&A: Chess board recognition system

Configuration (pip)

- Python3.9 [\[LINK\]](#)
- Add Python to `PATH` (environment variables)
- Install required packages
 - **For CPU users**
 - `pip install -r ./0-introduction/requirements_cpu.txt`
 - **For GPU users (please make sure CUDA 11.3 and cuDNN 8.2 are installed in your computer)**
 - `pip install -r ./0-introduction/requirements_gpu.txt`
- Please be aware of the dependencies if you have more than one Python installed
- VS Code [\[LINK\]](#) (optional)
 - Config Python3.9 in your VS Code

Configuration (Miniconda) (Recommended)

- Miniconda installation package download [\[LINK\]](#)

- Add Miniconda to `PATH` (environment variables)
- Open terminal in `vision2022/`
 - Create a virtual environment (replace `<environment name>` with a name given by you, e.g. `vision`)
 - `conda create -n <environment name> python=3.9`
 - Activate the virtual environment
 - `conda activate <environment name>`
 - Install required packages
 - **For CPU users**
 - `pip install -r ./0-introduction/requirements_cpu.txt`
 - **For GPU users** (`cuda-toolkit` will config CUDA 11.3 and cuDNN 8.2 automatically, so you don't need to install them manually)
 - `conda install pytorch==1.10.2 torchvision==0.11.3 torchaudio==0.10.2 cuda-toolkit=11.3 -c pytorch -c conda-forge`
 - `pip install -r ./0-introduction/requirements_gpu_conda.txt`

Test configuration

- Run `python ./0-introduction/show_a_cat.py` under `vision2022/`
 - Press `Esc` or `q` on the keyboard to close the window

```
1 import sys
2
3 print(sys.version)
```

```
1 3.9.16 (main, Mar 8 2023, 10:39:24) [MSC v.1916 64 bit (AMD64)]
```

```
1 import torch
2 import tensorflow as tf
3
4 print(torch.cuda.is_available())
5 # let's see the list of CUDA architectures, and the device name
6 if torch.cuda.is_available():
7     print(torch.cuda.get_device_name(device=None),
8           torch.cuda.get_arch_list())
9 print(tf.test.is_gpu_available())
10 # tf.test.is_gpu_available() is deprecated and will be removed soon
11 tf.config.list_physical_devices('GPU')
```

```
1 True
2 NVIDIA GeForce RTX 3060 Laptop GPU ['sm_37', 'sm_50', 'sm_60', 'sm_61',
3 'sm_70', 'sm_75', 'sm_80', 'sm_86', 'compute_37']
4 WARNING:tensorflow:From
5 C:\Users\lebro\AppData\Local\Temp\ipykernel_19472\519669526.py:8:
6 is_gpu_available (from tensorflow.python.framework.test_util) is deprecated
and will be removed in a future version.
7 Instructions for updating:
8 Use `tf.config.list_physical_devices('GPU')` instead.
```

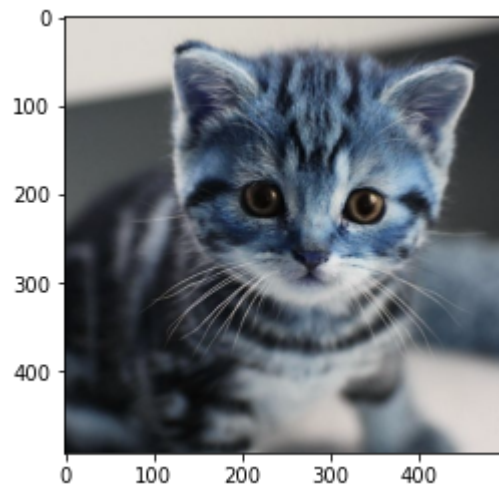
```
1 [PhysicalDevice(name='/physical_device:GPU:0', device_type='GPU')]
```

```
1 import cv2
2 import numpy as np
3 import matplotlib.colors as mat_color
4
5 print(cv2.__version__)
6
7 # read the cat image
8 path = "./images/cat.jpg"
9 img_bgr = cv2.imread(path)
10
11 print(type(img_bgr))
12 print(np.shape(img_bgr))
```

```
1 4.7.0
2 <class 'numpy.ndarray'>
3 (493, 493, 3)
```

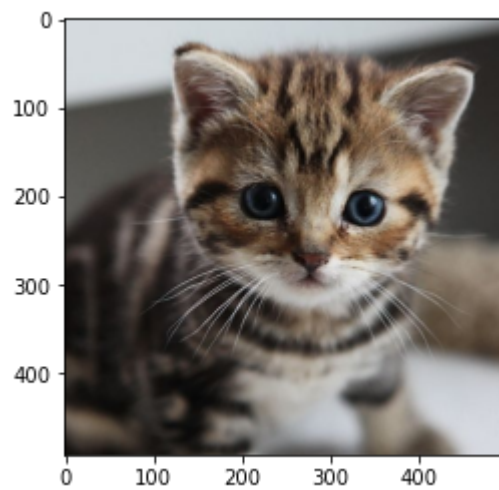
```
1 from matplotlib import pyplot as plt
2
3 # display img
4 no_norm = mat_color.Normalize(vmin=0, vmax=255, clip=False)
5 plt.imshow(img_bgr, norm=no_norm)
```

```
1 <matplotlib.image.AxesImage at 0x7ff9561ede80>
```



```
1 # bgr -> rgb
2 img_rgb = cv2.cvtColor(img_bgr, cv2.COLOR_BGR2RGB)
3 plt.imshow(img_rgb, norm=no_norm)
```

```
1 <matplotlib.image.AxesImage at 0x7ff954150040>
```



```
1 import os
2
3 # save the img in csv format
4 os.makedirs('../data', exist_ok=True)
5 data_file = os.path.join('../data', 'cat.csv')
6 print(data_file)
7 with open(data_file, 'w') as f:
8     f.write('R,G,B\n')
9     for row in img_rgb:
10         for rgb in row:
11             f.write(str(rgb[0]) + ',' + str(rgb[1]) + ',' + str(rgb[2]) +
'\n')
```

```
1 | ../data/cat.csv
```

The End

2022.3